Forecasting stock trends based on News' corpus by comparing three classifiers

YuPeng Liu
Department of Software Engineering, Harbin University of Science and Technology
flyeagle99@126.com

Gang Gao
Department of Software Engineering, Harbin University of Science and Technology
962181995@qq.com

Yun Chen
Department of Software Engineering, Harbin University of Science and Technology
100118037@qq.com

ABSTRACT

Stock price is usually influenced by various factors in the market, and usually shows non-linearity and uncertainty in price fluctuation. When we solve the stock price prediction problem, the effectiveness of different prediction methods are different either. So in order to choose the best method to solve this problem ,it is necessary to compare several better prediction methods which are chosen from many methods. There are classifier based on Naive Bayes, Maximum Entropy and Support Vector Machine respectively. Finally, we can get which on is the most efficient from the experiment result.

KEYWORDS

Stock price ,News corpus ,Prediction Method.

1 Introduction

Stock investment analysis is an indispensable part of stock investment and plays an extremely important role in the investment process. The stock market exhibits many different characteristics. Before entering the stock market, it must be clear about the risks, returns, instability, liquidity and timeliness of the stock.[1] Therefore, the forecast of stocks is also very significant. Doing so will not only better choose the right stock, but also better play the role of stock management.Using traditional predictive models is not very effective. The accuracy is low and the model is too simple, but for stock forecasting, this is an extremely complex forecasting model. Therefore, traditional forecasting methods are somewhat weak in stock forecasting. At this time, using a neural network algorithm is a good choice.Stock prediction through neural networks involves a variety of methods, such as NB, ME, SVM and so on[2]. In this paper, these three methods are used to analyze the short-term trend of stock prices based on news corpus.

NB(Naive Bayes classifier):In machine learning[3], naive Bayes classifiers are a family of simple "probabilistic classifiers"[4] based on applying Bayes' theorem with strong (naive) independence assumptions between the features. [5]Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression[6], which takes linear time, rather than by expensive iterative approximation[7] as used for many other types of classifiers.The discussion so far has derived the independent feature model, that is, the naive Bayes probability model. The naive Bayes classifier combines this model with a decision rule. One common rule is to pick the hypothesis that is most probable; this is known as the maximum a posteriori or MAP decision rule. The corresponding classifier, a Bayes classifier, is the function that assigns a class label for $\hat{y} = C_k$ some k as follows:

$$\hat{y} = \arg\max_{k \in \{1,\ldots,K\}} p(C_k) \prod_{i=1}^{n} p(x_i | C_k) \quad (1)$$

ME(Maximum Entropy):The central idea of the principle of maximum entropy is to make the entropy function the largest on the basis of the known information[8]. At this time, the random variable is the most uncertain, so that the choice we make is more objective than other choices. It doesn't need to know other constraints that we don't know in known information. If the principle of maximum entropy[9] is transformed into a mathematical programming problem, the maximum value of the entropy function is taken as the objective function of the plan, and the known information that should be satisfied is used as the constraint of the plan[10]:

$$P_w(y|x) = \frac{1}{Z_w(x)} \exp \sum_{i=1}^{n} w_i f_i(x, y) \quad (2)$$

$$Z_w(x) = \sum_{y} \exp(\sum_{i=1}^{n} w_i f_i(x, y)) \quad (3)$$

SVM(Support Vector Machine) :It is supervised learning models with associated learning algorithms [11]that analyze data used for classification and regression analysis[12] .Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling [13] exist to use SVM in a probabilistic classification setting). A SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as

possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

$$\max_{a \geq 0} W(a) = \max_{a \geq 0} (\sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i,j}^{n} a_i a_j y_i y_j x_j^T x_i), \quad (4)$$

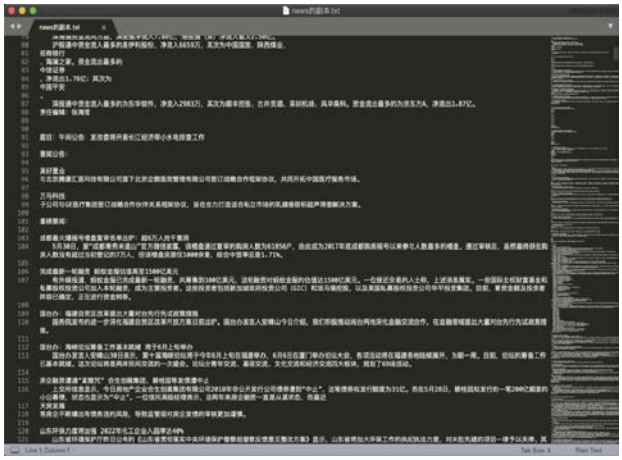
$$s.t., \sum_{i=1}^{n} a_i y_i = 0 \quad (5)$$

2 Goals and strategies

In this chapter, we describe the strategies and goals that have guided our method implementation.

There are three stages:First, we need to use these three methods to analyze News' corpus respectively ,then getting three results from the experiment. Finally, comparing these results and analyzing the quality of the text ,we are able to know which method is better than others ,which means that other researchers can utilize this method conveniently to save time and get high quality result.

3 Functional properties and implementation



.In this chapter, the properties of three methods would show in below part respectively. It is worth to mention that all of methods are based on the same News corpus (The Figure1 News' corpus)

Figure 1 News' corpus

The first method that we would describe is ME. In this way, we got the change of each precision by Iteration at first. (The Figure 2 Iteration method)Then , according to the polarity of single sentence in the corpus to analyze and get the training results.

The Figure 2 Iteration method

The second method is SVM , it is directly based on the Scikit-Learn machine library which is a free software machine learning library for the Python programming language. It features

```
def test_svm(self):
    print("SVMClassifier")
    print("——" * 45)
    print("Train num = %s" % self.train_num)
    print("Test num = %s" % self.test_num)
    print("C = %s" % self.C)

    from spa.classifiers import SVMClassifier
    svm = SVMClassifier(self.train_data, self.train_labels, self.best_words, self.C)

    classify_labels = []
    print("SVMClassifier is testing ...")
    for data in self.test_data:
        classify_labels.append(svm.classify(data))
    print("SVMClassifier tests over.")

    filepath = "f_runout/SVM-%s-train-%d-test-%d-f-%d-C-%d-%s-lin.xls" % \
               (self.type,
                self.train_num, self.test_num,
                self.feature_num, self.C,
                datetime.datetime.now().strftime(
                    "%Y-%m-%d-%H-%M-%S"))

    self.write(filepath, classify_labels, 2)
```

various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.[15]

```
def test_bayes(self):
    print("BayesClassifier")
    print("——" * 45)
    print("Train num = %s" % self.train_num)
    print("Test num = %s" % self.test_num)

    from spa.classifiers import BayesClassifier
    bayes = BayesClassifier(self.train_data, self.train_labels, self.best_words)

    classify_labels = []
    print("BayesClassifier is testing ...")
    for data in self.test_data:
        classify_labels.append(bayes.classify(data))
    print("BayesClassifier tests over.")

    filepath = "f_runout/Bayes-%s-train-%d-test-%d-f-%d-%s.xls" % \
               (self.type,
                self.train_num, self.test_num, self.feature_num,
                datetime.datetime.now().strftime(
                    "%Y-%m-%d-%H-%M-%S"))

    self.write(filepath, classify_labels, 0)

def write(self, filepath, classify_labels, i=-1):
    results = get_accuracy(self.test_labels, classify_labels, self.parameters)
    if i >= 0:
        self.precisions[i][0] = results[10][1] / 100
        self.precisions[i][1] = results[7][1] / 100

    Write2File.write_contents(filepath, results)
```

The Figure 3 SVM method

The third method is NB, and it just based on the classification algorithm directly. It is easy to use this function in our project.

The Figure 4 NB method

4.Experimental results and discussion

In this chapter ,we create a project include 5 main documents ,which are test.py, corpus.py, feature_extraction.py, tools.py, classifiers.py respectively.The classifiers.py document contains the methods previously mentioned above. Moreover, the corpus.py is the News' corpus document.All of the methods should be declared in the test.py, which means that it is the main function.What is more, it is important to describe that the

feature_extraction.py ,which use a statistical method to clear some Stop words for example'the', 'is' ,'at', 'which', 'on' and so on and leave some Emotional words.(The Figure 5 The statistical method)

```
@staticmethod
def _calculate(n_ii, n_ix, n_xi, n_xx):
    n_ii = n_ii
    n_io = n_xi - n_ii
    n_oi = n_ix - n_ii
    n_oo = n_xx - n_ii - n_oi - n_io
    return n_xx * (float((n_ii*n_oo - n_io*n_oi)**2) /
                   ((n_ii + n_io) * (n_ii + n_oi) * (n_io + n_oo) * (n_oi + n_oo)))

def best_words(self, num, need_score=False):
    words = sorted(self.words.items(), key=lambda word_pair: word_pair[1], reverse=True)
    if need_score:
        return [word for word in words[:num]]
    else:
        return [word[0] for word in words[:num]]
```

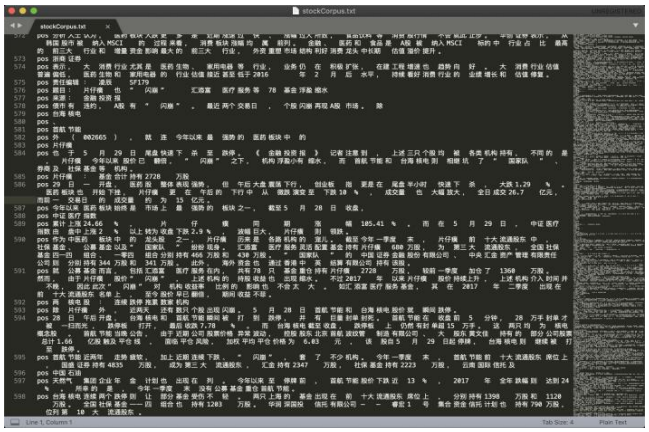
The Figure 5 The function in tools.py

Last but not least , the function of tools.py is significant either, which add the value of precision and recall to the result which are analyzed by these prediction methods.And we mainly separate two groups neg and pos for the corpus ,neg is the negative result(which means that the prospect of the corresponding share is poor) and the pos is the positive result(which means that the corresponding share is promising), and give the precision and recall to pos and neg ,and get the total of precision and recall, train number, test number and feature number.

```
xls_contents.extend([("neg-right", neg_right), ("neg-false", neg_false)])
xls_contents.extend([("pos-right", pos_right), ("pos-false", pos_false)])

pos_precision = pos_right / (pos_right + pos_false) * 100
pos_recall = pos_right / (pos_right + neg_false) * 100
pos_f1 = 2 * pos_precision * pos_recall / (pos_precision + pos_recall)
xls_contents.extend([("pos-precision", pos_precision), ("pos-recall", pos_recall), ("pos-f1", pos_f1)])

neg_precision = neg_right / (neg_right + neg_false) * 100
neg_recall = neg_right / (neg_right + pos_false) * 100
neg_f1 = 2 * neg_precision * neg_recall / (neg_precision + neg_recall)
xls_contents.extend([("neg-precision", neg_precision), ("neg-recall", neg_recall), ("neg-f1", neg_f1)])

total_recall = (neg_right + pos_right) / (neg_right + neg_false + pos_right + pos_false) * 100
xls_contents.append(("total-recall", total_recall))

print("    pos-right\tpos-false\tneg-right\tneg-false\tpos-precision\tpos-recall\t"
      "pos-f1\tneg-precision\tneg-recall\tneg-f1\ttotal-recall")
print("    " + "-----" * 45)
print("    %8d\t%8d\t%8d\t%8d\t%8.4f\t%8.4f\t%8.4f\t%8.4f\t%8.4f\t%8.4f\t%8.4f" %
      (pos_right, pos_false, neg_right, neg_false, pos_precision, pos_recall,
       pos_f1, neg_precision, neg_recall, neg_f1, total_recall))

return xls_contents
```

The Figure 6 The function in tools.py

Finally, we have separated the pos and neg for the corpus in Figure 8 and in Figure 9 respectively by using three methods and we also have got three different methods from the experiment .



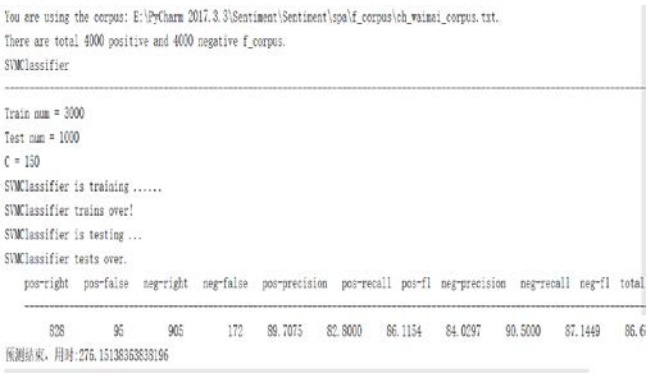
The Figure 8 Pos results

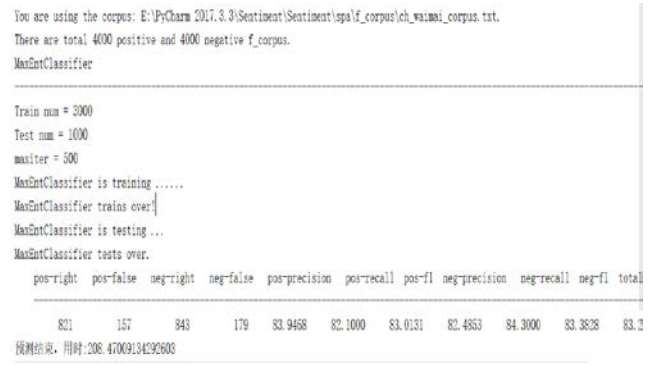
The Figure 8 Pos results

```
You are using the corpus: E:\PyCharm 2017.3.3\Sentiment\Sentiment\spa\f_corpus\ch_waimai_corpus.txt
There are total 4000 positive and 4000 negative f_corpus.
BayesClassifier
--------------------------------------------------------------------------------
Train num = 3000
Test num = 1000
BayesClassifier is training ......
BayesClassifier trains over!
BayesClassifier is testing ...
BayesClassifier tests over.
    pos-right   pos-false   neg-right   neg-false   pos-precision   pos-recall  pos-f1  neg-precision   neg-recall  neg-f1  total
    ----------------------------------------------------------------------------------------------------------------------------
          673          84         916         327         88.9036      67.3000     76.6079     73.6927     91.6000     81.6763     79.4
预测结束，用时:44.76959990562439

Process finished with exit code 0
```

It is obvious that these results can be seen in below Figures.

The Figure 10 The results of SVM





The Figure 11 The results of NB

The Figure 12 The results of ME

And we got the conclusion: In general, SVM is the most accurate but it cost most time by comparing with other ways. On the contrary, NB is the quickest than others because it not include any operation.The accuracy and spend of ME is in the middle of them.And we make several tables to show these classifiers' characteristics.

SVM	Accurate result		
Predicting result		Pos	Neg
	Pos	828	172
	Neg	95	105

Table 1 The Confusion Matrix of SVM

NB	Accurate result		
Predicting result		Pos	Neg
	Pos	821	843
	Neg	157	179

Table 2 The Confusion Matrix of NB

ME	Accurate result		
Predicting result		Pos	Neg
	Pos	673	327
	Neg	84	916

Table 3 The Confusion Matrix of ME

	Pos-precision	Neg-precision	Pos-f1	Neg-f1	Neg-recall	Pos-recall	Time
SVM	89.71	84.03	86.12	87.14	90.50	82.80	276.15
NB	88.90	73.69	76.61	81.68	91.61	67.30	44.77
ME	83.95	82.95	83.01	83.38	84.30	82.10	208.47

Table 4 The comparison of three classifiers

5 Conclusions and future work

With the rapid development of people's living standard, people tend to pay attention to stocks. Therefore, some people focus on using high-tech, such as Machine learning, to help them to find the right way to earn money from share market according to the advent of Neural Network. In this paper, we have tested three methods based on the News' corpus ,and got the result. For different utilization, people can choose corresponding method.We hope our experiment can give some helps for researchers who want to study this field, and plan to make more comparisons for other methods in Neural Networks in the future.

REFERENCES

[1] Wang Yu Chun, Application of Neural Network in Stock Forecasting. Communication world,(2019, 01).

[2] Russell, Stuart; Norvig, Peter (2003). Artificial Intelligence: A Modern Approach (2nd ed.). Prentice Hall. ISBN 978-0137903955.

[3] Yao Yu Qi, Research on Stock Analysis and Forecasting Model Based on Machine Learning.Market weekly(2019,02)

[4]]Ren Di, Wan Jian; Research on Web Service Quality Prediction Method Based on Bayesian Classification. Journal of Zhejiang University (Engineering Edition)(2017,06)

[5] Guo Xun Cheng, Application Research of Naive Bayes Classification Algorithm.Communication world,(2019,01,14)

[6] Holton, Glyn. "Numerical Solution, Closed-Form Solution".(2012,12).

[7] Amritkar, Amit; de Sturler, Eric; Świrydowicz, Katarzyna; Tafti, Danesh; Ahuja, Kapil (2015)

[8] Zhang Yao, Research on Continuous Maximum Entropy Stock Price Forecast Model Considering Transaction Cost. Liaoning University of Science and Technology(2015,03)

[9] Fang Ai Ping, Jia Yi ;Application of Maximum Entropy Principle in Probability Distribution Prediction. Physics and engineering(2017,12)

[10] Zhang Wen Da; Prediction model based on maximum entropy principle.Journal of Yuxi Teachers College(2017,02)

[11] Yang Jan Feng ,Wang Ning; A Survey of Machine Learning Classification Problems and Algorithms.Statistics and Decision(2019,03)

[12] Gao Wen,Research on stock intelligent investment strategy based on support vector machine parameter optimization algorithm.Shanghai Normal University,(2018,05)

[13] Lin, Hsuan-Tien; Lin, Chih-Jen; Weng, Ruby C. (2007)

[14] Zhang Yan You; Analysis of the principle of support vector machine.Shandong Industrial Technology.(2016,08)

[15] Fabian Pedregosa; Gaël Varoquaux; Alexandre Gramfort; Vincent Michel; Bertrand Thirion; Olivier Grisel; Mathieu Blondel; Peter Prettenhofer; Ron Weiss; Vincent Dubourg; Jake Vanderplas; Alexandre Passos; David Cournapeau; Matthieu Perrot; Édouard Duchesnay (2011)